Математика и Математическое моделирование
Сетевое издание
МГТУ им. Н.Э. Баумана
http://mathmjournal.ru

Ссылка на статью:
// Математика и Математическое моделирование. МГТУ им. Н.Э. Баумана. Электрон. журн. 2017. № 03. С. 64–76.

DOI: **10.24108/mathm.0317.0000077**

Представлена в редакцию: 05.05.2017
Исправлена: 19.05.2017

УДК 004.91

Применение метода ветвей и границ при решении задачи нечеткого поиска методом хеширования по сигнатуре в локальных базах данных

Хруничев Р.В.[1,*]

[*]hrunichev_robert@mail.ru

[1]Рязанский государственный радиотехнический университет, Рязань, Россия

В статье автор анализирует применение метода хеширования по сигнатуре при поиске в локальных базах данных. Указаны недостатки метода при таком типе поиска. Обоснована возможность уменьшения размера индекса сигнатуры при использовании лингвистических и статистических методов обработки текста. Показана актуальность задачи сравнения индексов запроса и образа терма в базе. Основное внимание автор уделяет анализу существующих методов сравнения индексов, обосновывается невозможность их применения к задаче нечеткого поиска. Разработана целевая функция для метода ветвей и границ, которая позволяет решить поставленную задачу при наличии разницы в одном бите индекса запроса и терма. результаты работы являются теоретическими. Автор указывает, что необходимы дальнейшие исследования в этом направлении, с целью усовершенствования алгоритма.

Ключевые слова: хеширование по сигнатуре; уменьшение размера индекса; сравнение битовых последовательностей; поиск в локальных БД; метод ветвей и границ

Введение

Важность задачи эффективного информационного поиска уже на протяжении нескольких десятков лет не теряет своей актуальности, кроме того, в связи с нарастающими объемами накопленных данных она только повышается. Поиск становится основной формой доступа к информации, причем потребность в сокращении временных затрат на отыскание нужных данных выходит на первое место [1- 4].

Анализ существующей литературы показал, что проблема глобального поиска, решаемая и развиваемая такими компаниями, как Google, Yandex, Yahoo и др. находится на вершине научных исследований в данной области знаний, чего нельзя сказать о локальном поиске. Эксперты среди важнейших задач аналитико–синтетической обработки данных как средства информационного поиска выделяют индексирование документов и информационных запросов [5].

На данном этапе развития координатное индексирование [9] – основной метод организации поиска документов в системах разного уровня в силу простоты и быстродействия.

Сотрудник компании Яндекс Сметанин Н.И. в аналитическом обзоре [6] делает вывод о том, что при нечётком поиске алгоритм координатного индексирования "хеширование по сигнатуре" дает один из наилучших результатов с точки зрения временных затрат при поиске. Отметим, что в локальных базах данных (часто это обычный ПК, на котором хранится документация учреждения/подразделения) минимальные временные затраты при поиске, а также размер индекса являются главными характеристиками. Алгоритмы [6] индексирования анализируются только на временные затраты при поиске, в то время как задача размера индекса не освещается. Так, например, при хешировании по сигнатуре предполагается двухбайтовый формат индекса. Вероятно, на размер индекса алгоритмы не анализировались, поскольку предполагается глобальный поиск, где данный параметр не столь важен.

Основная цель работы – сократить временные и вычислительные затраты при поиске в локальных БД, посредством изменения размера индекса при хешировании, а также применением метода ветвей и границ, для которого разработать целевую функцию при заданном количестве ошибок (разнице в индексах запроса и образа терма).

В работе приведены теоретические выкладки по применению метода ветвей и границ для сравнения индексов. В пункте 1.1 приведено краткое описание метода хеширования по сигнатуре и его основные преимущества, а также обоснованы его недостатки при применении в локальных базах данных, содержащих предметную коллекцию документов. Далее в пункте 1.2 рассматриваются методы, применяя которые можно добиться сокращения размера индекса сигнатуры. Обоснование проблемы сравнения индексов при нечетком поиске, в пункте 1.3, доказывает актуальность проблемы уменьшения временных и вычислительных затрат при их обработке. В пункте 1.4, перечислены методы сравнения индексов и сделан вывод об их невозможном применении при нечетком поиске. В заключении основной части – пункте 1.5 – приведена функция предлагаемого алгоритма сравнения сумм индексов и показано, что применение метода ветвей и границ не только хорошо подходит для поставленной задачи, но и позволяет при правильном выборе параметров целевой функции добиться снижения вычислительных затрат.

1. Основная часть

Задача проводимого исследования – показать, что существующие алгоритмы, например, исключающего «ИЛИ» (побитовое сравнение), последовательного сравнения, регистра сдвига и др. не подходят для задачи нечеткого поиска, с точки зрения показателей вычислительных и временных затрат при реализации в локальных базах данных. Для метода ветвей и границ разработать и исследовать целую функцию на сравнение **индексов** терма запроса и его поискового образа при **одном допустимом несовпадении** бит**ов**.

1.1 Метод хеширования по сигнатуре

В качестве алгоритма для индексирования выделенных основ термов применим метод хеширования по сигнатуре, который обладает рядом преимуществ [10]:

- позволяет осуществлять с высоким быстродействием поиск на точное равенство и поиск, допускающий одну-две «ошибки» в задании поискового запроса;
- эффективен, как в случае «прямых» чтений с диска, так и из кэша;
- использует компактный индекс. При правильном выборе параметров объем индекса не более чем на 10-20% превышает размер файла, содержащего список терминов словаря;
- отличается простотой реализации.

Суть метода [10].Пусть задан непустой алфавит A и известны вероятности появления различных символов алфавита. Пусть также на множестве символов A задана функция $f(\alpha)$, отображающая буквы в числа от 1 до *m*. Эта функция, как несложно видеть, задает разбиение алфавита на *m* подмножеств.

Определение. Сигнатурой $sign(w)$ слова *w* будем называть вектор размерности *m*, *k* - ый элемент которого равняется единице, если в слове *w* есть символ α такой, что $f(\alpha)=k$, и нулю в противном случае. Номером сигнатуры слова будем называть число $H(w)=\sum_{i=0}^{m-1}2^i\cdot sign(w)_{i+1}$. $H(w)$ является хэш-функцией, отображающей множество слов в отрезок целых чисел от 0 до 2^m-1, что позволяет организовать словарь в виде хеш-таблицы.

Процесс вычисления хеша: каждому биту хеша сопоставляется группа символов из алфавита. Бит 1 на позиции *i* в хеше означает, что в исходном слове присутствует символ из *i*-ой группы алфавита. Порядок букв в слове абсолютно никакого значения не имеет [11].

В таком алгоритме для полного покрытия k ошибок нужно изменять не менее 2k бит в хеше. Время работы, в среднем, при k «неполных» (вставки, удаления и транспозиции, а также малая часть замен) ошибках: $O\left(|H|^k\cdot n/2^{|H|}\right)$.

Существующий алгоритм [6, 10, 11, 12] хеширования по сигнатуре при применении его в локальных базах данных обладает рядом недостатков:

1) обработка двухбайтовой сигнатуры требует значительных временных затрат;
2) распределение букв по группам в сигнатуре никак не обосновано (существуют битовые позиции, которые с большой долей вероятности принимают значение "0");
3) разница количества термов соответствующих различным сигнатурам велика;
4) при сравнении сигнатур терма запроса и поискового образа возможно большое количество совпадений.

1.2 Предлагаемая подход к сокращению размера индекса сигнатуры

Размер сигнатуры – одна из основных характеристик с точки зрения временных и вычислительных ресурсов на поиск при локальном поиске. Во-первых, необходимо хранить сам индекс и при добавлении новых документов в базу производить переиндексацию, что при больших размерах индекса приведет к значительным временным затратам. Во-вторых, при поиске происходит процесс вычисления значения хеш-функции и, чем больше индекс, тем больше вычислительные затраты.

Основной характеристикой локальных БД является их предметная ориентированность, поэтому применением словарей предметной области можно сократить число анализируемых термов и этим, сократить количество термов, приходящихся на одну сигнатуру.

В работах [6, 10, 11] рассматривается применение метода хеширования по сигнатуре при глобальном поиске, где применение предметных словарей не имеет смысла, поэтому количество термов на сигнатуру может быть велико. С этой точки зрения двухбайтовый формат сигнатуры обоснован, поскольку очень большое число термов на сигнатуру влияет на качество поиска. Однако, распределение символов языкового алфавита по группам в сигнатурах никак не обосновано – просто последовательное распределение. Очевидно, что, например, бит, соответствующий редко встречающимся символам «Щ» и «Ъ», в большинстве сигнатур будет принимать значение «0», т.е. хранение такого бита нецелесообразно.

При анализе локальных базах данных число участвующих в анализе термов достаточно просто сократить, применяя, например, статистические методы обработки (частотный анализ текста и закон Ципфа) [7, 8], предметные словари и лингвистическую обработку текстов. Таким образом, можно осуществить переход от двухбайтового формата к однобайтовому. Это может привести к увеличению числа термов, приходящихся на одну сигнатуру, но, поскольку поиск осуществляется в локальной базе, то их число не будет велико.

1.3. Потребность в сравнения индексов при нечетком поиске

Предварительно обозначим задачу необходимости разработки целевой функции для метода ветвей и границ с возможными изменениями в одной битопозиции (при одном пропуске/вставке, замене символов, приводящей к изменению одного бита). Ранее было показано, что уменьшение размера сигнатуры до однобайтового формата может привести к появлению нескольких термов – подмассив общего количества термов, выделенных в процессе анализа документов всей коллекции локальной базы данных, – соответствующих одной сигнатуре. Значение хеш-функции $H(w_i)$, в случае безошибочного запроса, является ключом к тому подмассиву индексов термов, которые соответствуют одной сигнатуре $sign(w_i)$. В случае, если сигнатуре $sign(w_i)$ соответствует один терм, то разработка целевой функции не требуется – полученный терм удовлетворяет задаче поиска. Но если же, подмассив термов включает более одного элемента (в литературе употребляют термин

"коллизия"), то уже возникает необходимость решения задачи обработки индексов термов образов массива/подмассива на предмет соответствия индексу терма запроса.

Также отметим, что задача нечеткого поиска основана на предположении о возможном допущении ошибок (пропуске буквы или наборе лишней, замена символов) в запросе. Перестановка букв в терме не приводит к изменению сигнатуры. Но, при *возможном* допущении ошибки возможны следующие гипотезы:

1) изменения индекса не произошло (в запросе ещё есть символы алфавита из данной группы), что не привело к изменению сигнатуры и по ключу $H(w_i)$ найден подмассив, в котором один терм образ – соответствие установлено;
2) изменения индекса не произошло, что не привело к изменению сигнатуры и по ключу $H(w_i)$ найден подмассив, в котором несколько термов образов; требуется сравнить все индексы термов образов подмассива на соответствие индексу запроса;
3) произошло изменение одного бита сигнатуры и ключу $H(w_i)$ соответствует пустая стока в базе (при анализе коллекции документов не было термов, соответствующих данному ключу) и в этом случае нужно просмотреть все индексы массива на соответствие индексу запроса или выводить сообщение об отсутствии документов по запросу;
4) произошло изменение одного бита сигнатуры, но алгоритм "не понимает", что совершена ошибка и вычисленному сигнатурному ключу $H(w_i)$ находится соответствие – подмассив, в котором нет термов, соответствующих запросу, даже с учетом возможного изменения в одном бите терма запроса; в этом случае нужно просмотреть всю базу индексов на соответствие индекса запросу;
5) произошло изменение одного бита сигнатуры, но алгоритм "не понимает", что совершена ошибка и вычисленному сигнатурному ключу $H(w_i)$ находится соответствие – подмассив, в котором есть термы, соответствующие запросу, с учетом того, что возможно изменение в одном бите терма запроса; в этом случае нужно просмотреть подмассив индексов на соответствие индекса запросу.

Как видим, в четырех из пяти возможных случаев, придется осуществлять сравнения индексов. Здесь и становится актуальной задача разработки оптимальной целевой функции, позволяющей сократить временные и вычислительные затраты на сравнение индексов образа терма в базе и терма запроса.

1.4 Существующих методы побитового сравнения индексов

В аналитической статье [13] автор рассматривает задачу сравнения битовых последовательностей. Рассматриваются стандартные подходы к решению задачи сравнения, например, последовательное ("шаблонное") сравнение и регистр сдвига, но здесь же делается вывод о сложности реализации и больших временных затратах при реализации.

В качестве одного из возможных решений предлагается построение конечного автомата, но он не позволяет реализовать задачу нечеткого поиска, т.е. не допускает "мягкого" поиска.

Также в литературе **изложены** методы побитового исключающего "ИЛИ", побитового "И", побитового "ИЛИ", побитовое "НЕ", но эти методы также основан на принципе точного соответствия, и "мягкий" поиск не реализуют [14].

1.5 Предлагаемое решение задачи нечеткого поиска при возможном изменении одного бита индекса запроса

Решение задачи разработки целевой функции возникает при реализации любой из четырех гипотез, указанных в предыдущем пункте. Основная задача формулируется следующим образом: *для метода ветвей и границ разработать целевую функцию, позволяющую решить задачу нечёткого поиска с учетом возможного изменения в одном бите* (допущении ошибки в запросе, которая привела к изменению индекса). Сразу оговоримся, что при большем числе изменений в битах запроса задача становится существенно сложнее и автором не ставится цель ее решения. В этом случае возможно существенное снижение качества поиска и целесообразность исследования ставится под вопрос.

Ключ $H(w_i)$ обращает поисковый алгоритм к подмассиву, в котором хранятся индексы образов термов, выделенных в процессе анализа, или же возникает задача сравнения всех индексов массива образов термов, выделенных в процессе анализа. Индексы термов могут быть получены, например, также по сигнатурному методу, где каждому символу алфавита соответствует один бит или любому другому методу, реализацией которого является индекс терма. Задача получения индексных последовательностей термов, выделенных в процессе анализа базы документов, автором не ставится. Здесь могут быть проведены дополнительные исследования на размер индекса и его возможного уменьшения, но данная задача не является предметом рассмотрения в данной статье.

Итак, приведем реализацию целевой функции для метода ветвей и границ, применительно к задаче нечеткого поиска и поясним суть ее применения.

Целевая функция:

$$f(x)=\left|\sum_{i=1}^{m}x_i-\sum_{i=1}^{m}x_i'\right| \underset{x\in D}{\leq} 1, \qquad (1)$$

при условии, что x принадлежит множеству, которое задано ограничениями

$$D:\begin{cases}\sum\limits_{j=1}^{n}\left(\sum\limits_{\substack{k=3j-2,\\ k\leq m}}^{3j}x_{jk}-\sum\limits_{\substack{k=3j-2,\\ k\leq m}}^{3j}x'_{jk}\right)\leq 1,\\ x\in\{0,1\}.\end{cases} \qquad (2)$$

В формулах (1) и (2) m – размер индекса терма запроса и образа в базе, n=div(m, 3)+1 – количество выделенных непересекающихся троек (триад) бит в индексе (+1 в слу-

чае некратного трем числа бит в индексе), x_i и x_i' – значение i–го бита в индексах терма запроса и образа, j – номер битовой триады в индексе, k – позиция в триаде. Очевидно, что сравнивая индексы, переменная x может принимать значения «0» или «1». Значение функции вычисляется по модулю, поскольку при ошибке **(**в виде вставке лишнего символа**)** сумма единиц в индексе запроса может быть больше, чем сумма единиц в индексе образа.

Суть предлагаемого алгоритма. Поскольку задача нечеткого поиска решается с допущением о том, что возможно изменение в одно**м** бит**е**, то ограничения меньше единицы показывают, что разность сумм индексов не должна превышать этого порогового значения. Но, поскольку размеры индексов термов могут быть большими (для задачи качественного поиска), то вычислять разницу сумм для каждой пары индексов запроса и образа нерационально. По всем m позициям временные и, главное, вычислительные затраты будут значительными.

Предлагается для каждой пары индексов запроса и образа вычислять промежуточные разницы сумм по трем битам (**2**). Тогда, если эта разница больше порога «1» для любой тройки, центрального 3j-1 и соседних 3j-2 и 3j бит, то соответствие считается не установленным и происходит переход к сравнению индексов запроса и следующего индекса образа в массиве/подмассиве. Очевидно, что для большинства индексов в массиве/подмассиве разность троек бит не будет удовлетворять условию целевой функции на множестве D гораздо быстрее, чем вычисление разности сумм по всем m элементам индексов запроса и образа.

В качестве оценок на условие ограничения на множестве D были выбраны триады по двум причинам. Во-первых, при использовании двух бит для ограничения условие бы не выполнялось только в 2 случаях из возможных 16 – при сравнении «00-11» и «11-00». Это дает вероятность невыполнения условия равную 0,125 на каждую двойку, следовательно, происходит большее количество сравнений, что ведет к увеличению вычислительных затрат. Во-вторых, при ограничении в 4 и более бит, вероятность невыполнения условия растет уже заметно медленнее, и, например, для 4 бит сравнения составляет ≈0,29 на каждую четверку, но при этом происходит значительное увеличение затрат на вычисление сумм. По предварительной оценке оптимальное значение на невыполнение условия по вероятностным (≈0,22 на тройку) и вычислительным характеристикам достигается для ограничения триадами. Возможно разделение четверками даст более лучший результат, но здесь необходим ряд экспериментов на временные затраты по вычислению сумм, что также, пока, не являлось целью исследования.

Приведем рисунок, помогающий понять работу алгоритма.

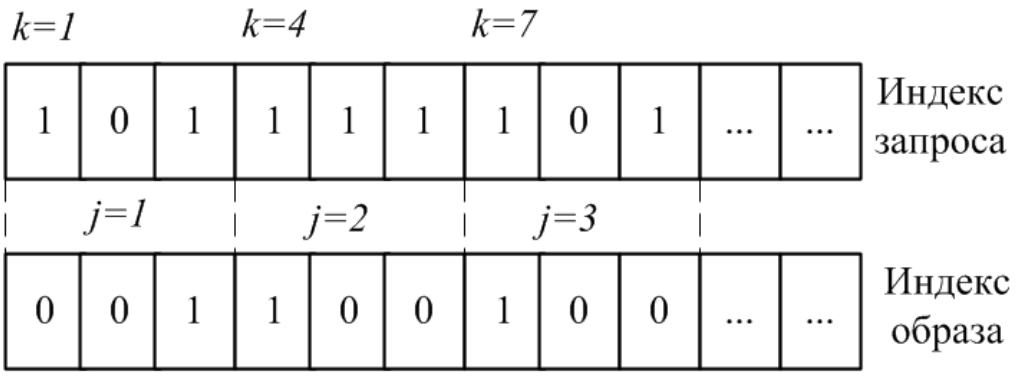

Рис. 1. Сравнение индексных последовательностей запроса и образа триадами.

На приведенном рисунке видно, что, например, для первой триады (j=1) условия по ограничению выполняются – $\sum_{k=1}^{3} x_k - \sum_{k=1}^{3} x'_k \le 1$, а для второй тройки ($j$=2) уже нет – $\sum_{k=4}^{6} x_k - \sum_{k=4}^{6} x'_k > 1$. В этом случае осуществляется переход к следующей ветви общего дерева индексов термов в массиве/подмассиве. Таким образом, осуществив предварительную сортировку индексов в подмассивах, можно значительно ускорить процесс отыскания нужного терма при сравнении суммами по трем битам. Также видим, что если хотя бы в одной тройке бит, за исключением последней, условие ≤ 1 не выполнится, то это уже приведет к сокращению вычислительных и временных затрат.

Заключение

По результатам проведенного исследования можно сделать следующие выводы:

- при поиске в локальных базах возможно осуществить переход от двухбайтового к однобайтовому формату при получении сигнатуры терма;
- при неравномерном распределении количества термов на сигнатуру показана актуальность задача сравнения индексных последовательностей термов запроса и образа в базе;
- для осуществления нечеткого поиска с возможной одной ошибкой в бите запроса предложен метод ветвей и границ с разработанной целевой функцией и набором ограничений;
- применяемые методы позволяют сократить вычислительные и временные затраты при реализации задачи нечеткого поиска в локальных базах данных.

При анализе предложенного алгоритма можно сказать, что он как минимум не приведет к увеличению вычислительных и временных затрат – тот случай, когда сравнение ведется до последней триады включительно, а как максимум даст значительное сокращение указанных параметров – когда уже первая триада дает разницу сумм больше 1. Задача поиска в локальных базах данных зачастую решается исследователями-одиночками, поскольку коммерческий эффект минимален, чего нельзя сказать о глобальном поиске, где

сосредоточены передовые лаборатории мира. Но это не означает, что работы в этом направлении не ведутся – задача по-прежнему актуальна и будет более актуальной в будущем, поскольку объем данных постоянно растёт.

В качестве задач на дальнейшее исследование автор ставит:

1) изучить методы для возможного усреднения количества термов, соответствующих одной сигнатуре, например, учитывая частоту появления символов в алфавите;
2) рассмотреть целесообразность задачи сравнения индексов с учетом двух возможных изменениях в индексах;
3) для одного изменения в индексе сравнить вычислительные и временные затраты при сравнении методом ветвей и границ по разнице сумм троек и четверок бит.

Список литературы

1. Национальный корпус русского языка: Сайт. Режим доступа: http://www.ruscorpora.ru/index.html (дата обращения 06.07.2017).
2. Маннинг К.Д., Прабхакар Рагхаван, Шютце Х. Введение в информационный поиск: пер. с англ. М.: Вильямс, 2011. 520 с. [Manning Ch.D., Prabhakar Raghavan, Schutze H. Introduction to information retrieval. Camb.; N.Y.: Camb. Univ. Press, 2008. 482 p.].
3. Цукерт А.Г. Проблемы и перспективы информационного поиска // Изв. Таганрог. гос. радиотехн. ун-та. 2001. Т. 21. № 3(21). С. 194–201.
4. Задачи поисковых систем. Режим доступа: http://asknet.ru/Technology/searchtask.htm (дата обращения 06.07.2017).
5. Зупарова Л.Б., Зайцева Т.А. Аналитико-синтетическая переработка информации: учебник. М.: ФАИР, 2008. 400 с.
6. Сметанин Н. Нечёткий поиск в тексте и словаре. Режим доступа: https://habrahabr.ru/post/114997/ (дата обращения 06.07.2017).
7. Хруничев Р.В. Принципы построения многомерного пространства терминов в процессе анализа предметно-ориентированной коллекции документов // Вестник Астрахан. гос. техн. ун-та Сер.: Управление, вычислительная техника и информатика. 2012. № 1. С. 136-141.
8. Zipf G.K. Selected studies of the principle of relative frequency in language. Camb.: Harvard Univ. Press, 1932.
9. Гиляревский Р.С. Основы информатики: Курс лекций. М.: Экзамен, 2003. 318 с.
10. Мосалев П.М. Обзор методов нечеткого поиска текстовой информации // Вестник Моск. гос. ун-та печати им. И. Федорова. 2013. № 2. С. 87-91.
11. Нгуен Ной Хыу. Обзор некоторых алгоритмов нестрогого сопоставления записей применительно к задаче исключения дублирования персональных данных // Молодой ученый. 2013. № 5. С. 163-166.
12. Бойцов Л.М. Поиск по сходству в документальных базах данных: хеширование по сигнатуре оптимальное соотношение скорости поиска, простоты реализации и объема

индексного файла [текст] // 8-я Междунар. конф. «Математика. Компьютер. Образование»: МКО 2001 (Москва, 2-4 октября 2001 г.): Труды. М.: Прогресс-Традиция, 2001. С. 177-183.

13. Панкратов И.В. Одновременный поиск нескольких двоичных шаблонов в потоке с помощью конечного автомата // Прикладная дискретная математика. 2014. № 2(24). С. 119–125.
14. Побитовые операторы. Режим доступа: https://learn.javascript.ru/bitwise operators (дата обращения 06.07.2017).

Mathematics & Mathematical Modelling
Electronic journal of the Bauman MSTU
http://mathmjournal.ru

Mathematics and Mathematical Madelling of the Bauman MSTU, 2017, no. 03, pp. 64–76.

DOI: **10.24108/mathm.0317.0000077**

Received: 05.05.2017
Revised: 19.05.2017

Method of Branches and Boundaries to Solve a Fuzzy Search Problem by Hash Signature Method in Local Databases

R.V. Khrunichev[1,*]

[*]hrunichev_robert@mail.ru

[1]Ryazan State Radio Engineering University, Ryazan, Russia

Keywords: hash signature, reducing the size of the index, comparing punch-out sequences, to search the local database, the method of branches and boundaries

The article highlights a relevant problem of fuzzy information search in local databases. A choice of the hash signature method was based on the existing analysis of coordinate indexing methods for global search. When searching, it provides the least time characteristics.

Analysis of using this method in local databases has shown that there are a number of shortcomings such as a large size of the signature index, an unjustified distribution of the alphabet symbols in groups, the uneven number of terms corresponding to one signature.

Some of the shortcomings were eliminated in the course of research activities. It is shown that to eliminate the drawback of a large size of the signature index the linguistic and statistical methods of text processing can be used. This allows you to reduce the number of terms involved in the analysis, reduce the size of the index and the number of terms that correspond to one signature.

A task of the fuzzy search is to search for possible errors in the query. The article draws special attention to the analysis of possible situations when there is a difference in indices of the query and the image of the term in the database in one bit. Based on the results of the theoretical analysis, it was concluded that when conducting the fuzzy search a problem of comparing bit sequences often arises. The currently implemented algorithms for bitwise index comparisons do not consider this type of tasks. In this connection, a target function to compare indices has been developed on the basis of the branch and boundary method. Note that the problem of obtaining bit sequences of terms was not considered. A theoretical analysis of the target function has shown that the optimal time and computational costs are attainable for a given set. A constraint (boundary) has been developed. It allows us reduce time costs and in many cases not compare indices of the query and the image terms completely. All the steps in the study were aimed at reducing the time and computational costs, which in the conditions of local search is relevant.

In conclusion, it was noted that the target function developed by the branch and boundary method can be further investigated for the application of constraints. Also, as a direction for

studying, a possible probabilistic analysis of the distribution of terms in groups for the hash signature method was chosen.

References

1. *Natsionalnyj korpus russkogo iazyka* [National corpus of the Russian language]: Site. Available at: http://www.ruscorpora.ru/index.html , accessed 06.07.2017 (in Russian).
2. Manning Ch.D., Prabhakar Raghavan, Schütze H. *Introduction to information retrieval*. Camb.; N.Y.: Camb. Univ. Press, 2008. 482 p. [Russ. ed.: Manning Ch.D., Prabhakar Raghavan, Schütze H. *Vvedenie v informatsionnyj poisk*. Moscow: Williams Publ., 2011. 520 p].
3. Zukert A.G. Problems and perspectives of information retrieval. *Izvestiia Taganrogskogo gosudarstvennogo radiotekhnicheskogo universiteta* [Izvestiia of the Taganrog State Radiotechnical Univ.], 2001, no. 3(21), pp. 194-201 (in Russian).
4. *Zadachi poiskovykh sistem* [Tasks search engines]. Available at: http://asknet.ru/Technology/searchtask.htm, accessed 06.07.2017 (in Russian).
5. Zuparova L.B., Zaitseva T.A. *Analitiko-sinteticheskaia pererabotka informatsii* [Analytic-synthetic processing of information]: a textbook. Moscow: Fair Publ., 2008. 400 p. (in Russian).
6. Smetanin N. *Nechetkij poisk v tekste i slovare* [Fuzzy search in text and dictionary]. Available at: https://habrahabr.ru/post/114997/, accessed 06.07.2017 (in Russian).
7. Khrunichev R.V. Principles of construction of the multidimensional space of terms in the process of analyzing a subject-oriented collection of documents. *Vestnik Astrakhanskogo gosudarstvennogo tekhnicheskogo univ. Ser.: Upravlenie, vychislitel'naia tekhnika i informatika* [Vestnik of Astrakhan State Technical Univ. Ser.: Management, Computer Engineering and Informatics], 2012, no. 1, pp. 136-141 p. (in Russian).
8. Zipf G.K. Selected studies of the principle of relative frequency in language. Camb.: Harvard Univ. Press, 1932.
9. Giliarevskyj R.S. *Osnovy informatiki: Kurs lektsij* [Foundations of computer science: A course of lectures]. Moscow: Ekzamen Publ., 2003. 318 p. (in Russian).
10. Mosalev P.M. Review of methods for fuzzy searching of textual information. *Vestnik Moskovskogo gosudarstvennogo universiteta pechati im. I. Fedorova* [Vestnik MGUP by Ivan Fedrorov], 2013, no. 2, 87-91 (in Russian).
11. Nguyen N.H. Overview of some algorithms for fuzzy matching records with respect to the task to avoid duplication of personal data. *Molodoj uchenyj* [Young Scientist], 2013, no. 5, pp. 163-166 (in Russian).
12. Bojtsov L.M. Poisk po skhodstvu v dokumentalnykh bazakh dannykh: kheshirovanie po signature – optimal'noe sootnoshenie skorosti poiska, prostoty realizatsii i ob'ema indeksnogo fajla [Similarity search in documentary databases: hashing by signature optimum ratio of the search speed, simplicity of implementation and the volume index file]. *8-ia Mezhdunarodnaia konferentsiia "Matematika. Komp'yuter. Obrazovanie": MKO 2001* [8th Intern. Conf. "Mathe-

matics. Computer. Education" (Moscow, October 2-4, 2001)]: Proc. Moscow: Progress-Traditsiia Publ., 2001. Pp. 177-183 (in Russian).

13. Pankratov I.V. Simultaneous search for several binary patterns in a stream with finite-state automation. *Prikladnaia diskretnaia matematika* [Applied Discrete Mathematics], 2014, no. 2(24), pp.119-125 (in Russian).
14. *Pobitovye operatory* [The bitwise operators]. Available at: https://learn.javascript.ru/bitwise-operators , accessed 06.07.2017 (in Russian).